Real-time Vehicle Motion Estimation Using Texture Learning and Monocular Vision

Yann Dumortier, Mikäel Kais and Rodrigo Benenson
IMARA project team, INRIA
B.P 105 78153 Le Chesnay Cedex France
Email: {firstname}.{lastname}@inria.fr

***Abstract*— High integrity localization system is an important challenge to improve safety for road vehicles. A way to meet the requirements is to fuse information from several sensors, from position and orientation sensors to motion, speed and acceleration sensors. This paper tackles the problem of vehicle motion estimation using monocular vision. A geometric model of the road is used to learn a texture patch in the current image, this patch is then tracked through the successive frames to estimate in real time the motion of the vehicle. The proposed method was assessed using a centimeter accuracy Real Time Kinematic GPS receiver.**

I. Introduction

A. Context

There is a wide variety of initiatives trying to address the problem of road safety. In this context, the European Commission has an ambitious goal of traffic death reduction of 50% by 2010. Some approaches relie on the delivery of alerts and assistance to drivers since human factor is the primary cause of fatalities. In such Advance Driver Assistance Systems, a high integrity localization system is a requierement.

The localization process is usually based on fusion of exteroceptive information and proprioceptive information. While exteroceptive measurements provide position and/or orientation of the mobile with respect to a reference frame, usually by matching range and angle measurement with a model which can be static or even dynamic if considering the Global Positioning System (GPS), proprioceptive information provides motion, speed and acceleration information that has to be integrated with time.

However, availability of satellite-based positioning system such as GPS receivers is not guaranteed since the signals from satellites can easily be affected in given environments due to the so called canyon effect of simply multipaths that can affect electromagnetic waves [2]. It is thus of importance to fuse GPS data with other information to continue the estimation process in case of GPS outage.

Laser scanners [12], [13] may be used to compute position and orientation if an a priori model exists or is being built. Motion may also be estimated by matching static points from one laser scan to another. Inertial Measurement Units (IMU) and odometry provide an alternative way to estimate the motion of a mobile. However, laser scanners need salient features to extract useful informations and both laser sensors and precise IMU are expensive.

Camera is a low cost alternative sensor that may be used to estimate the motion between two frames. Classical methods may be divided in two groups: those using no a priori model and the others. The model can be a pattern collection or a 3D Computer Aided Design (CAD) model [11] . Methods that do not require a priori model allow to ride through new environments but requires however at first a feature learning stage. Textures, as used in [9] with a visual Simultaneous Localization And Mapping (SLAM) algorithm, or geometric features, such as edges [10], [14], may also be employed with monocular vision or stereo vision head [4].

B. Model description

Since road environments are very heterogeneous, our method focuses only on the road itself. Nevertheless, its relief is not salient enough to provide reliable positioning information and its boundaries do not make it possible to extract depth translation [1]. But we can easily assume that roads always have useful texture considering that recent roads have white markers whereas older ones show sufficient wear and tear marks. So, we assume the road locally flat and choose an approach based on monocular vision.

Because it performs on a limited number of small planar patches centered on points of interest, without any constraint between them, the SLAM method as proposed in [9] is not adapted to track only one large plane. Therefore we decide to design a localization approach, suitable in a large number of cases, since it is only based on natural road landmarks. Our method proposes, using a geometric a priori model of the road, to learn the texture, then to track it in the following frames.

The approach presented in this paper is shown Fig. 1. At first, a 3D CAD model is projected in the image plane, the resulting patch is mapped with its texture during the learning stage. Hypothesis on the camera motion is used to compute several transformation matrices that are applied iteratively to warp the patch. At each iteration, a correlation is performed between the warped patch and the current frame patch. From

the warp allowing the optimal match, we get the inverse vehicle motion in the local reference frame used.

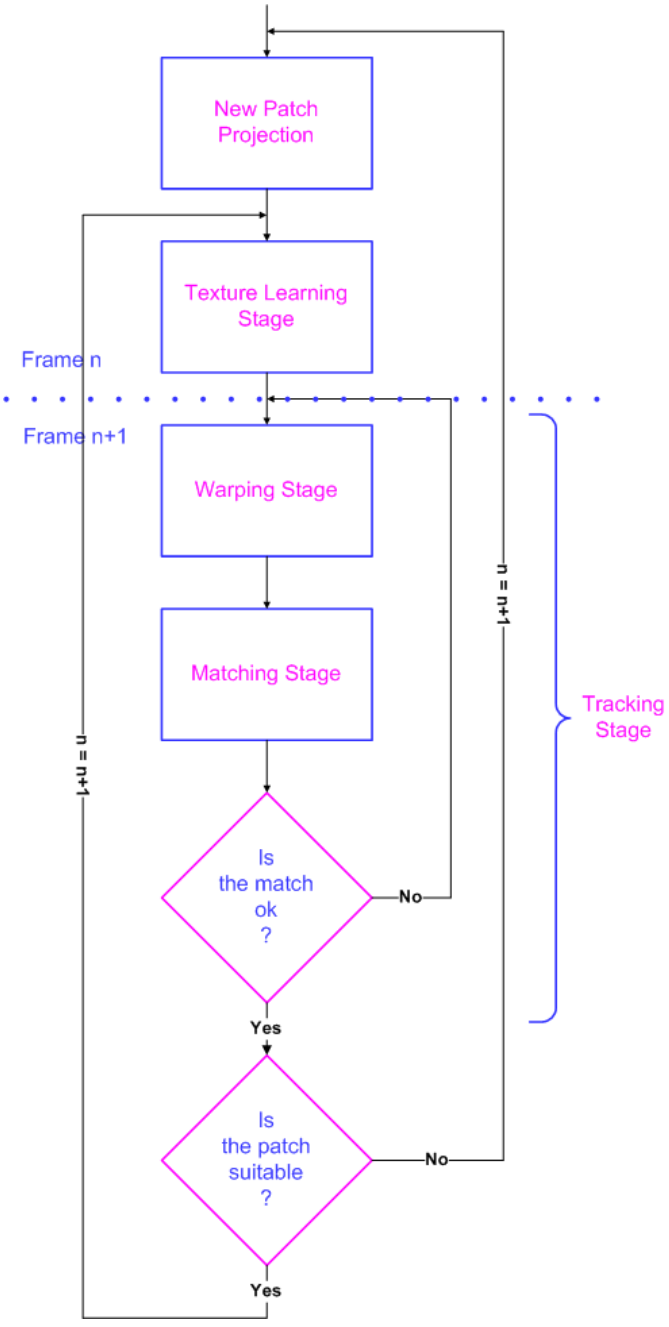

Fig. 1. Diagram of the proposed algorithm

The paper is organized as follows, section II presents the camera model used and the mapping process. Section III deals with the planar homography and the warping stage. In sections IV and V we discuss how to perform a fast patch correlation, a brief overview of the global motion computation is also provided. Experimental results and assessment are presented section VI. At last, our conclusions and directions for future works are given in section VII.

II. Texture Learning

The learning stage extracts a patch texture from the current image using the pinhole camera model assumption and the intrinsic parameters associated.

Let $\mathbf{P}$ be a point in the camera referential system and $\mathbf{p}$ its correspondent in the image plane $\mathcal{P}^2$ with respectively $[x\ y\ z\ 1]^T$ and $[u\ v\ 1]^T$ their homogeneous coordinates, so:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{z} \underbrace{\begin{bmatrix} fk_u & 0 & u_0 & 0 \\ 0 & fk_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}}_{\mathrm{K}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \qquad (1)$$

where the intrinsic parameters are f the effective focal length, $[u_0\ v_0]^T$ the image coordinates of the intersection of the optical axis of the lens with the image plane, k_u and k_v the scale factors. All these parameters are obtained by offline calibration.

The texture is directly mapped on the CAD patch projection using current frame pixels (Fig. 2). At each new frame, the texture is updated instead of always working on its first instance. In such manner, and assuming that the vehicle moves forward, the accuracy is improved in spite of a weak drift which may be introduced by successive projections. When a part of the tracked patch is no more in the field of view, a new available one is projected and mapped further.

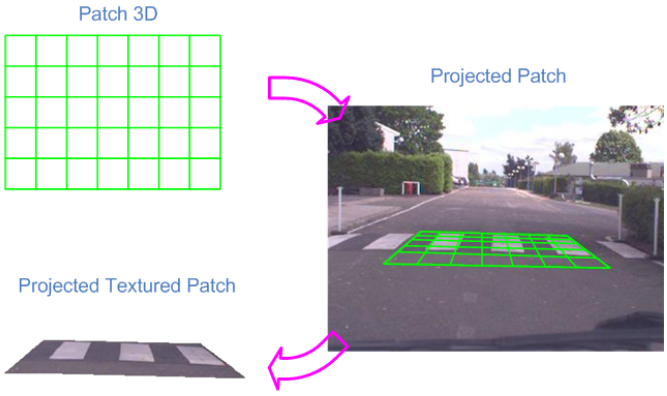

Fig. 2. Texture learning stage

III. Texture Warping

The tracking stage computes the 3D transformation which matches the patch extracted from the previous frame in the current one. Since the road is assumed to be locally plane, the transformation is a warp between two planar patches describing the camera 3D motion and projection. Such a perspective projection restricted to coplanar points can be expressed as an homography which is a projective transformation between two planes [3].

Let us consider the 3D planar patch projection into a camera image plane. The homography describing this transformation from $\mathcal{R}^3$ to the projective plane $\mathcal{P}^2$ is:

$$\mathbf{p} = \mathsf{H}_{IR}\,\mathbf{P}\ ,$$

with $\mathbf{P}$ a point of the patch in the scene, $\mathbf{p}$ its projection onto the image plane and $\mathsf{H}_{\mathcal{IR}}$ the homography warping from $\mathcal{R}^3$ to $\mathcal{P}^2$. Homographic transformations being bijective, the relationship between two projections of the same world point in two camera planes, differing only by their point of view, can be expressed by:

$$\mathbf{p}_2 = \mathsf{H}_{\mathcal{IR}}\,\mathcal{T}\,\mathsf{H}_{\mathcal{IR}}^{-1}\,\mathbf{p}_1\ ,$$

where $\mathcal{T}$ is a 4×4 matrix describing the camera rigid displacement in $\mathcal{R}^3$. Therefore the homographic matrix from the video frame i_1 to i_2 has the form:

$$\mathsf{H}_{21} = \mathsf{H}_{\mathcal{IR}}\,\mathcal{T}\,\mathsf{H}_{\mathcal{IR}}^{-1}\ . \tag{2}$$

Since the homography matrix $\mathsf{H}_{\mathcal{IR}}$ is in fact the projection matrix K including the intrinsic parameters, (2) becomes:

$$\mathsf{H}_{21} = \mathsf{K}\,\mathcal{T}\,\mathsf{K}^{-1}\ .$$

Let C_1 and C_2 be two cameras looking at the same plane π. We note $\mathbf{P_i}$ a point of π in the camera C_i coordinate system, and $\mathbf{p}_i$ its projection in the corresponding image plane (Fig. 3).

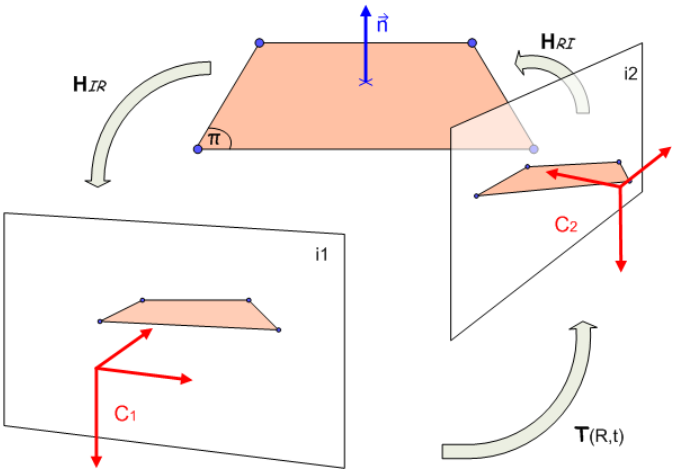

Fig. 3. Homography between two camera projections

The equation describing the change of referential is:

$$\mathbf{P}_2 = \mathsf{R}\,\mathbf{P}_1 + \mathbf{t}\ , \tag{3}$$

where R and $\mathbf{t}$ are respectively the rotation matrix and the translation vector which describe the referential change. Let $\mathbf{n}$ be a unit vector normal to the plane π such as:

$$\mathbf{n}^T\,\mathbf{P}_i = d \qquad ,\forall i \in \{1,2\}\ ,$$

with d the distance between the camera and the plane, so:

$$\frac{\mathbf{n}^T}{d}\,\mathbf{P}_i = 1 \qquad ,\forall i \in \{1,2\}\ ,$$

therefore, (3) becomes:

$$\mathbf{P}_2 = \mathsf{R}\,\mathbf{P}_1 + \frac{\mathbf{t}\,\mathbf{n}^T}{d}\,\mathbf{P}_1\ . \tag{4}$$

The planar homography between the two image planes includes the camera projection matrix (K_1 and K_2 respectively to the camera C_1 and C_2) :

$$\mathbf{p}_2 = \underbrace{\mathsf{K}_2\,(\mathsf{R} + \frac{\mathbf{t}\,\mathbf{n}^T}{d})\,\mathsf{K}_1^{-1}}_{\mathsf{H}_{21}}\,\mathbf{p}_1\ .$$

Our model uses only one moving camera. Consequently, $\mathcal{T}_{(\mathsf{R},\mathbf{t})}$ describes the camera motion, and we assume there is only one constant projection matrix K. Therefore :

$$\mathsf{H}_{21} = \mathsf{K}\,(\mathsf{R} + \frac{\mathbf{t}\,\mathbf{n}^T}{d})\,\mathsf{K}^{-1}\ . \tag{5}$$

IV. Texture Matching

A. The cost function

The matching stage attends to assess each warping. In order to respect real-time system constraint, we perform a fast patch correlation by computing the sum of absolute values differences (SAD) for each pixel of the patch. The SAD is performed for each red, green and blue image component and the best warp minimizes the following cost function :

$$f_{cost} = \frac{1}{3N}\sum_{i=1}^{N}\sum_{j=1}^{3}(\left|P_{patch(ij)} - P_{image(ij)}\right|)\ , \tag{6}$$

where N is the pixel number, $P_{patch(i)}$ the i^{th} pixel value of the warped patch and $P_{image(i)}$ its correspondent in the real image. j says the color component.

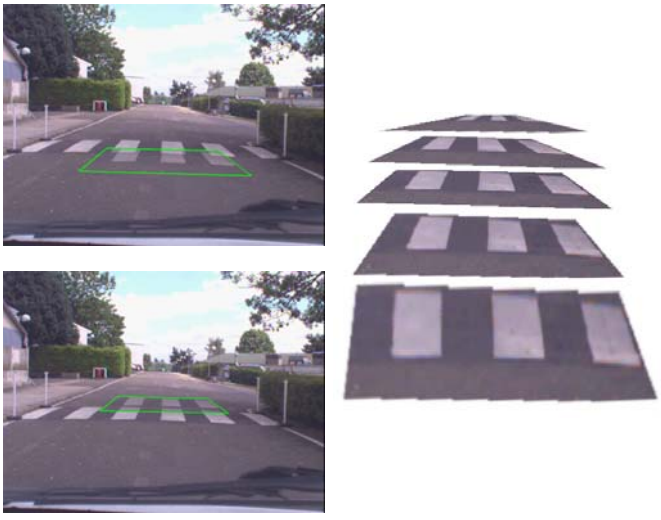

Fig. 4. Matching stage illustration

Performing the correlation between the two patches in $\mathcal{R}^3$ could have introduced imprecision due to the interpolation process. Therefore, the SAD is computed in the image plane between the predicted patch projection and the current image (Fig. 4).

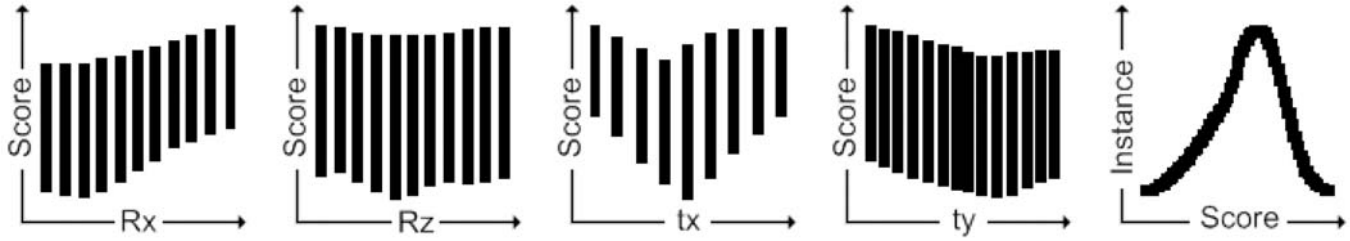

Fig. 5. Cost function profile close to the true motion. Four degrees of freedom are described (from the left to the right: pitching and cap rotations, then lateral and depth translations)

B. Tracking method

An exhaustive study of the cost function have been performed in the warping space around the optimal transformation. Fig. 5 shows an example of such an analysis in the solution neighborhood using Ggobi, a visualization software for viewing high dimensional data. One can observe cost function profiles contingent on pitching, cap, lateral and depth translations. Histogram representation allows to describe the score amplitude due to the other parameters. The best score is clearly highlighted since there is no local minima that appears in a the area close to the global minima score.

The camera frame rate is supposed high enough to use the hypothesis of a vehicle with constant speed. Consequently, the previous motion is used as the initialization guess for the current motion estimation. At the start of the algorithm, the assumption of a standing vehicle is used. Assuming such approach avoid the algorithm being trapped in a local minima and a gradient descent algorithm is used to find the optimal solution in a very short time.

Considering the six Degrees Of Freedom (DOF), we note the start point of the optimization process $\mathbf{X}_1 = (Rx_1, Ry_1, Rz_1, tx_1, ty_1, tz_1)$, therefore a gradient descent step is performed such as:

$$\mathbf{X}_{i+1} = \mathbf{X}_i + \Delta X \ ,$$

with:

$$\Delta X = -\eta \, \nabla f(X) \ ,$$

where ∇f is the cost function gradient and η the training step. The cost function gradient may be expressed as the sum of partial derivative of the function with respect to the vector X as following:

$$\nabla f(X) = \frac{\partial f}{\partial Rx}(X) + \frac{\partial f}{\partial Ry}(X) + \cdots + \frac{\partial f}{\partial tz}(X) \ .$$

Several tests using time integrated IMU data as first motion estimation have been performed without any improvement. Such results confirm that the assumption of vehicle constant speed and the assumption of no local minima close to the global minima were taken wisely.

V. Motion computation and local referential

The final path motion includes all successive warps extracted from the tracking stage as following:

$$\mathcal{T}_{n0} = \mathcal{T}_{n\,n-1} \cdots \mathcal{T}_{21} \ \mathcal{T}_{10} \ \ ,$$

considering $\mathcal{T}_{j\,i}$ the motion from the position i to the position j and all transformations expressed in a same absolute referential. Since in our case each tracking stage computes the patch motion in the current local referential system, we have first to change the referential as following.
Let the first local referential $\mathcal{R}_0$ to be our absolute referential and $\mathsf{T}_{\mathcal{R}_0\mathcal{R}_n}$ the transition matrix from $\mathcal{R}_n$ to $\mathcal{R}_0$ expressed by:

$$\mathsf{T}_{\mathcal{R}_0\mathcal{R}_n} = \mathsf{T}_{\mathcal{R}_0\mathcal{R}_1} \ \mathsf{T}_{\mathcal{R}_1\mathcal{R}_2} \cdots \mathsf{T}_{\mathcal{R}_{n-1}\mathcal{R}_n} \ ,$$

hence:

$$\mathcal{T}_{/\mathcal{R}_0} = \mathsf{T}_{\mathcal{R}_0\mathcal{R}_n} \ \mathcal{T}_{/\mathcal{R}_n} \ ,$$

where $\mathcal{T}_{/\mathcal{R}_n}$ is the motion $\mathcal{T}$ in the local coordinate system $\mathcal{R}_n$ associated.

VI. Experimentation and results

A. Trajectory estimation

Our approach was validated with the LARA INRIA car on several experiments with a duration of 30 seconds to 60 seconds. The extrinsic camera calibration was performed using the coplanar POSIT algorithm [7] on known road patterns such as a pedestrian crossing. In order to assess the imprecision of the localization, a centimeter accuracy Real Time Kinematic (RTK) GPS receiver was installed in the vehicle (Fig. 7). Graphics Fig. 6 represent the paths from a centimeter accuracy RTK GPS and integration of the motions given by assessed algorithm, respectively in green and red.

The algorithm estimates correctly the vehicle motion when the road texture is sufficient (see Fig. 8 and Fig. 9 to look at good and bad cases). The first graph Fig. 9 illustrates an insufficient texture case after 125 meters and we observe a lateral shift from this point.

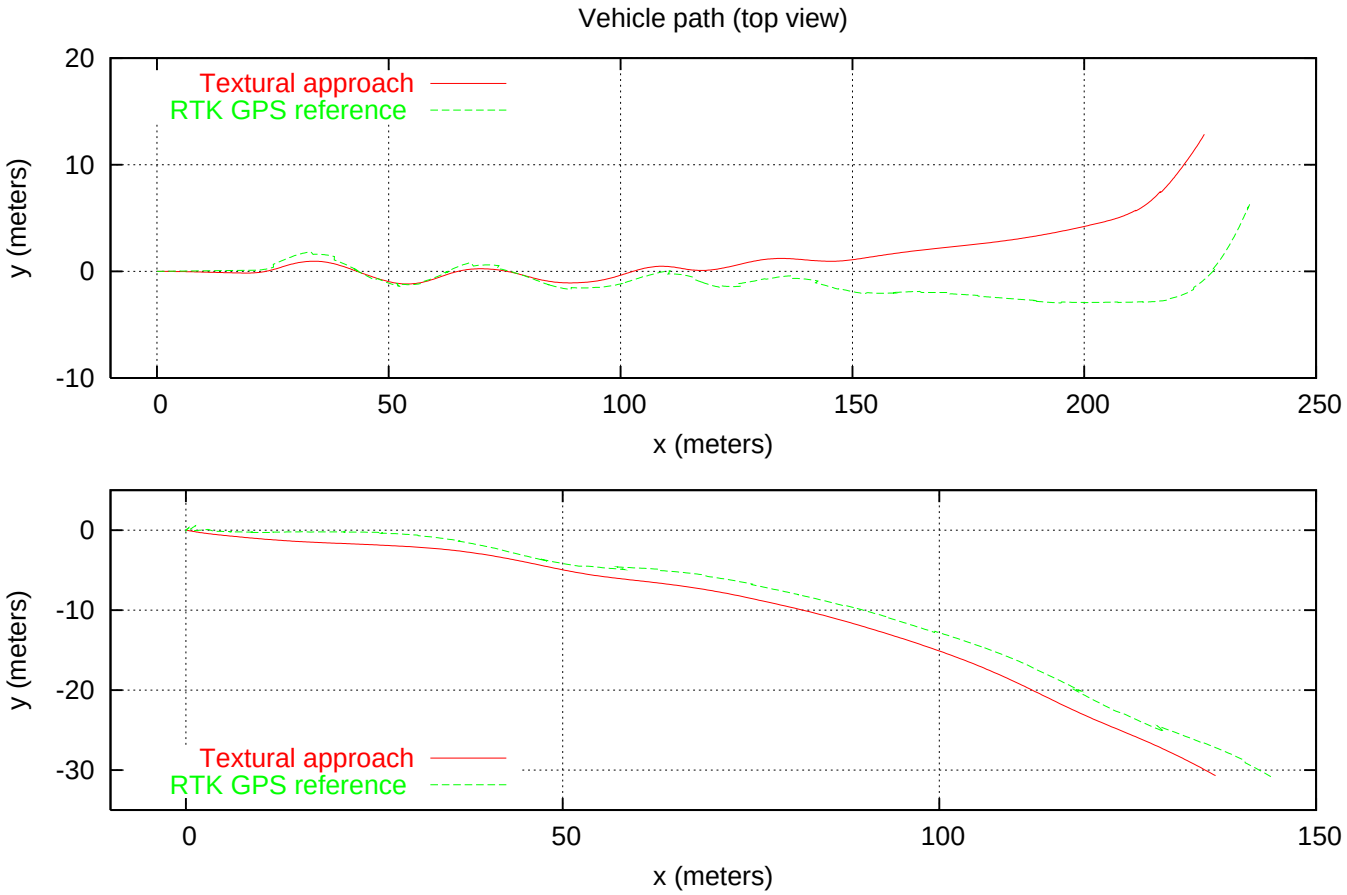

Fig. 6. Vehicle paths estimation by RTK GPS (green) and our visual perception approach (red)

B. Computational cost

The whole process, including learning texture, warping predicted patch by homography, matching it with the previous frame using a gradient descent and the real-time visualization is performed with a current-day PC (3 GHz) at a frequency of 15 Hz using color images with a resolution of 640×480 pixels.

Fig. 7. Instrumented vehicle

VII. Conclusion

In this paper, a real-time vehicle motion estimation system is described. The motion estimation is formulated as a matching task where the objective is to minimize a cost function. The approach is applicable to multi patch tracking. Indeed, it could be used to track simultaneously several planes, road plane, building planes... Performance of the vision motion estimator were assessed using a high accuracy Real Time Kinematic GPS receiver.

Future work will focus on adding a texture evaluation module coupled with the matching module to assess in real time the imprecision given by the motion estimator and output the imprecision associated with the motion measurement. This last module is a requirement before integrating the information from this real time motion estimation system in the Bayesian fusion framework for localization that was developed by our laboratory, it is also a step towards a high integrity localization system.

Acknowledgment

The authors wish to acknowledge Mr. André Ducrot (IMARA team of INRIA) and Mr. André Gagalowicz (MIRAGE team of INRIA) for their active contribution to this

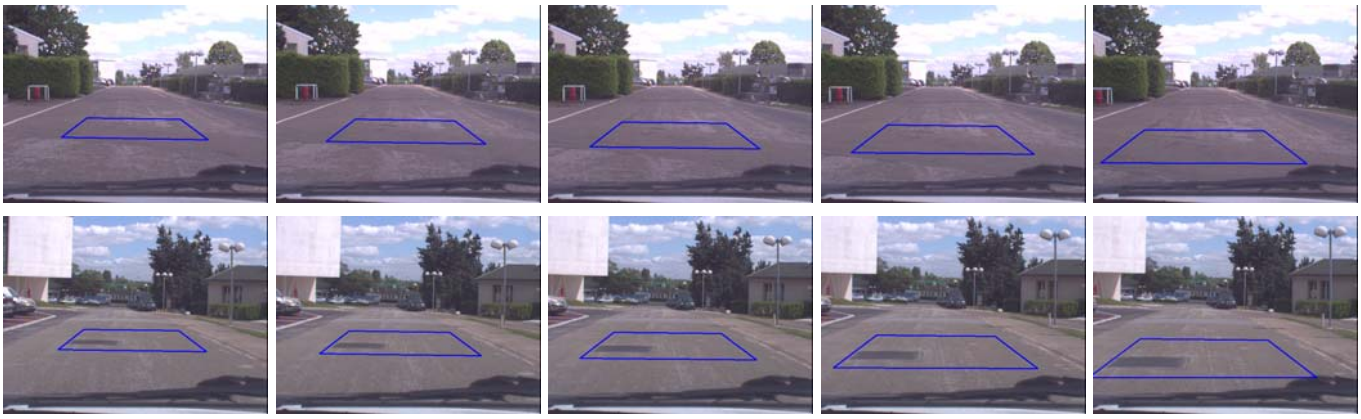

Fig. 8. Two correct patch tracking sequences (we can notice in the first one that very few texture information enable a good tracking)

Fig. 9. An insufficient texture area (the patch is shifting)

work. The authors would like to thank Mr. Nicolas Simond (INRIA-EMP Joint Research Unit) for fruitful discussions.

REFERENCES

[1] F. Chausse, V. Voisin, J. Laneurit and R. Chapuis, *Centimetric Localization of a Vehicle Combining Vision and Low Cost GPS* IAPR Workshop on Machine Vision Applications (MVA'02), Japan, December 2002

[2] J. Chao, Y. Chen, W. Chen, X Ding, Z Li, N Wong and M Yu, *An Experimental Investigation Into the Performance of GPS-based Vehicle Positioning in Very Dense Urban Areas* Journal of Geospatial Engineering, Vol. 3, No. 1, pp. 59-66, 2001

[3] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision* Cambridge University Press, 2000

[4] N. Simond and P. Rives, *Trajectography of an Uncalibrated Stereo Rig in Urban Environments* IEEE RSJ/International conference on Intelligent Robot and System (IROS'04), 2004

[5] S. Benhimame and E. Malis, *Real-time Image-based Tracking of Planes Using Efficient Second-order Minimization* IEEE RSJ/International conference on Intelligent Robot and System (IROS'04), 2004

[6] P. Gérard and A. Gagaglowicz, *Three Dimensional Model-based Tracking Using Texture Learning and Matching* Pattern Recognition Letters, 2000

[7] D. Oberkampf, D. F. DeMenthon and L. S. Davis, *Iterative Pose Estimation Using Coplanar Feature Points* Computer Vision and Image Understanding, vol. 63, no. 3, pp. 495.511, may 1996.

[8] F. Jurie and M. Dhome, *Real-time 3D Template Matching*, Computer Vision and Pattern Recognition, 2001

[9] A. J. Davidson, I. D. Reid and N. D. Molton, *Locally Planar Patch Features for Real-time Structure from Motion*, Proceeding of British Machine Vision Conference, 2004

[10] C. Doignon, G. Abba and E. Ostertag, *Recognition and Localization of Solid Objects by a Monocular Vision System for Robotic Tasks*, Intelligent Autonomous Systems, 1995

[11] T. Drummond and R. Cipolla, *Real-time Tracking with Complex On-line Camera Calibration*, Proceeding of British Machine Vision Conference, September 1999

[12] D. Haehnel, D. Fox, W. Burgard, and S. Thrun, *A Highly Efficient FastSLAM Algorithm for Generating Cyclic Maps of Large-scale Environments from Raw Laser Range Measurements*, Proceedings of the Conference on Intelligent Robots and Systems (IROS'03), 2003

[13] E. Ollivier and M. Parent, *Odometric Navigation with Matching of Landscape Features*, Singapore, ICARV 2002

[14] R. Sim and G. Dudek, *Mobile Robot Localization From Learned Landmarks*, Proceedings of the Conference on Intelligent Robots and Systems (IROS'98), 1998

[15] F. Caballero, L. Merino, J. Ferruz and A. Ollero, *A Visual Odometer Without 3D Reconstruction for Aerial Vehicles. Applications to Building Inspection*, Proceedings of International Conference on Robotics and Automation (ICRA'05), Spain, April 2005